Architecture Development and Validation

Presented at the JTRS Step 2 Industry Day

28 June 1999

CDR Jim Smith
JTRS Chief Engineer
703.588.1293
smithjd@sarda.army.mil
http://www.jtrs.sarda.army.mil

Overview

- Background
- Architecture Development
- Architecture Validation
- Summary

Background

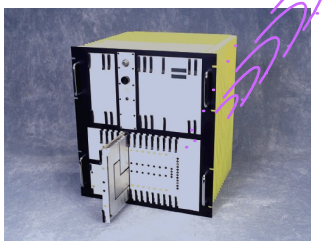
Architecture: The Bottom Line

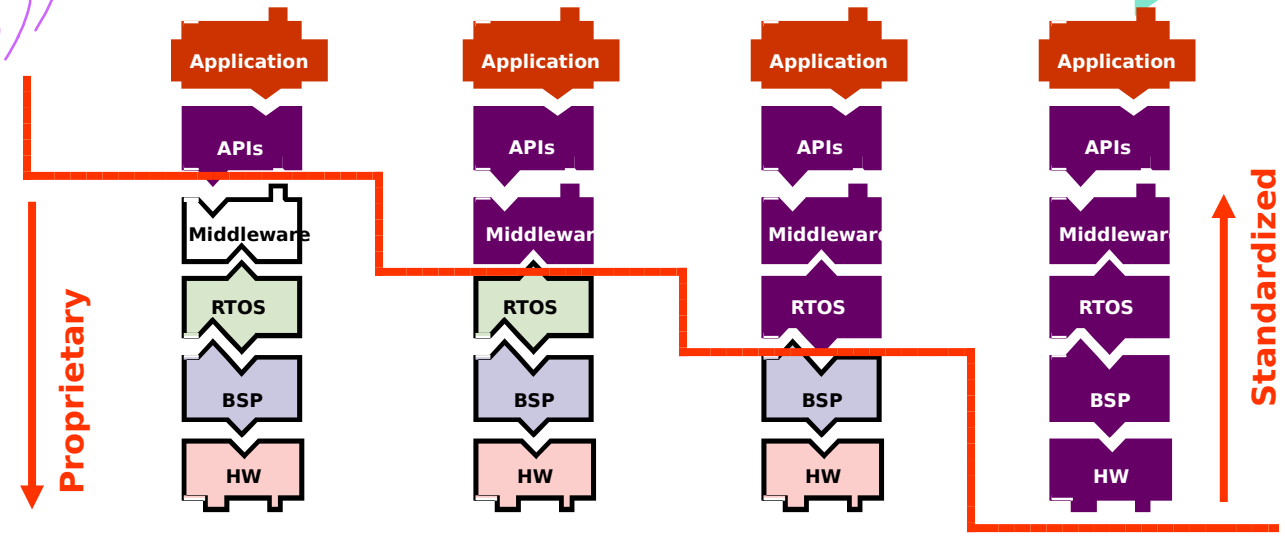
The Question: What Degree of Openness Delivers Systems That Are …

- Software Programmable and Reprogrammable
- Interoperable
- ORD Compliant
- Affordable
- Extensible
- Scalable
- and have Type 1 Security,

… AND …

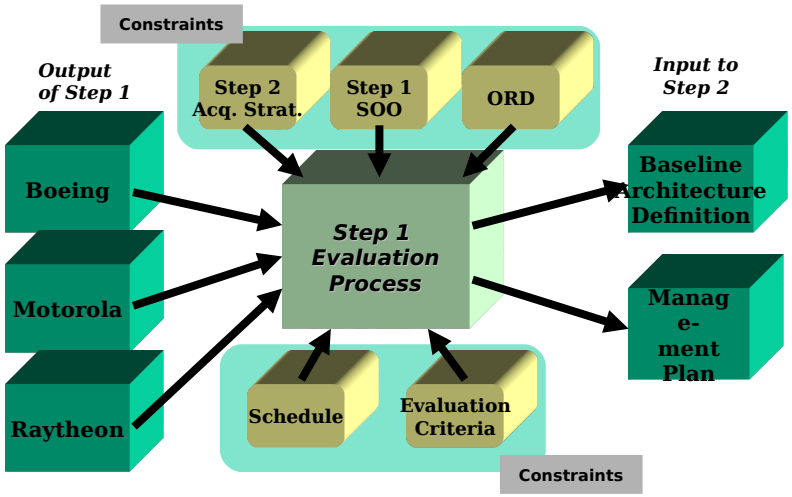
Proprietary Systems ← ***Where Do We "Draw the Line"?*** → **Completely DoD Standardized**





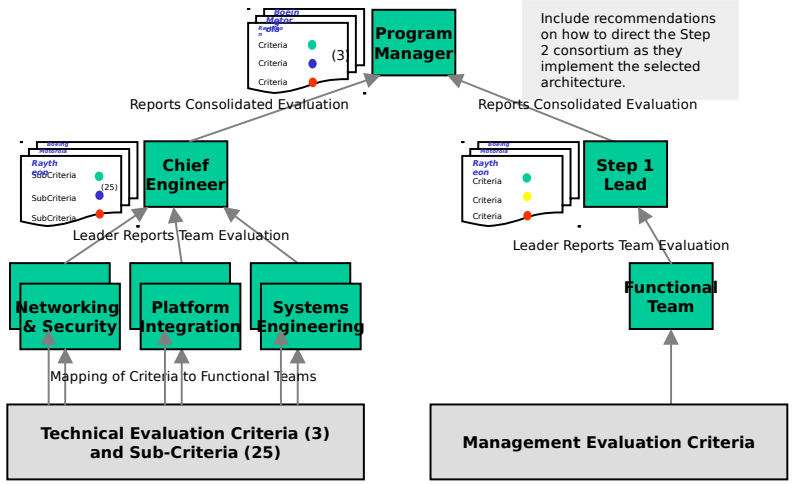
… Leverages Commercial Investments and Provide a Competitive Business Case?

Step 1 Evaluation Process



- Take Step 1 Results (i.e., Architecture Definition Reports (ADRs) and Management Plans) and Evaluate Them With Respect To Established Constraints (e.g., Cost, Schedule, and Performance Considerations)
- Results of Evaluation Formed the Basis for Step 2 Effort (Develop/Validate Architecture)

- Management and Technical Evaluations Conducted Separately
  - Seven Technical Functional Teams
  - Single Management Team
- Evaluators Drawn From Multiple Services/Agencies
- Outside Functional Area Experts (e.g., SEI, OSJTF) Used to Provide Independent Evaluation
- Evaluation Results Combined at PM-Level


Program Manager
Include recommendations on how to direct the Step 2 consortium as they implement the selected architecture.
Reports Consolidated Evaluation
Reports Consolidated Evaluation
Chief Engineer
Step 1 Lead
Leader Reports Team Evaluation
Leader Reports Team Evaluation
Networking & Security
Platform Integration
Systems Engineering
Functional Team
Mapping of Criteria to Functional Teams
Technical Evaluation Criteria (3) and Sub-Criteria (25)
Management Evaluation Criteria

Step 1 Technical Evaluation Process

Technical Evaluation Criteria

- **JTRS Baseline Architecture Definition**
  - Compliance with ORD Requirements
  - Interoperability
  - Programmability and Reprogrammability
  - Independence of SW and HW
  - Industry Acceptance as Commercial Standard
  - Extendibility to New Software Applications/Hardware Components
  - Scalability of Architecture
  - Affordability
  - INFOSEC (Architecture)
  - Identification of critical Interfaces
  - Performance
  - System Control
  - Modularity
  - Evolvability
  - Architecture Definition Level
- **Technology Issues**
  - Incorporate new technologies into JTRS
  - Technology Trends
  - Technology dependencies
  - Technology obsolescence
- **Implementation Issues**
  - Multi-band, multi-mode operation
  - Wideband operation, Antennas, power amplifiers and co-site interference
  - Networking issues
  - INFOSEC (Implementation)
  - Hardware and Software growth
  - Use of architecture across multiple domains

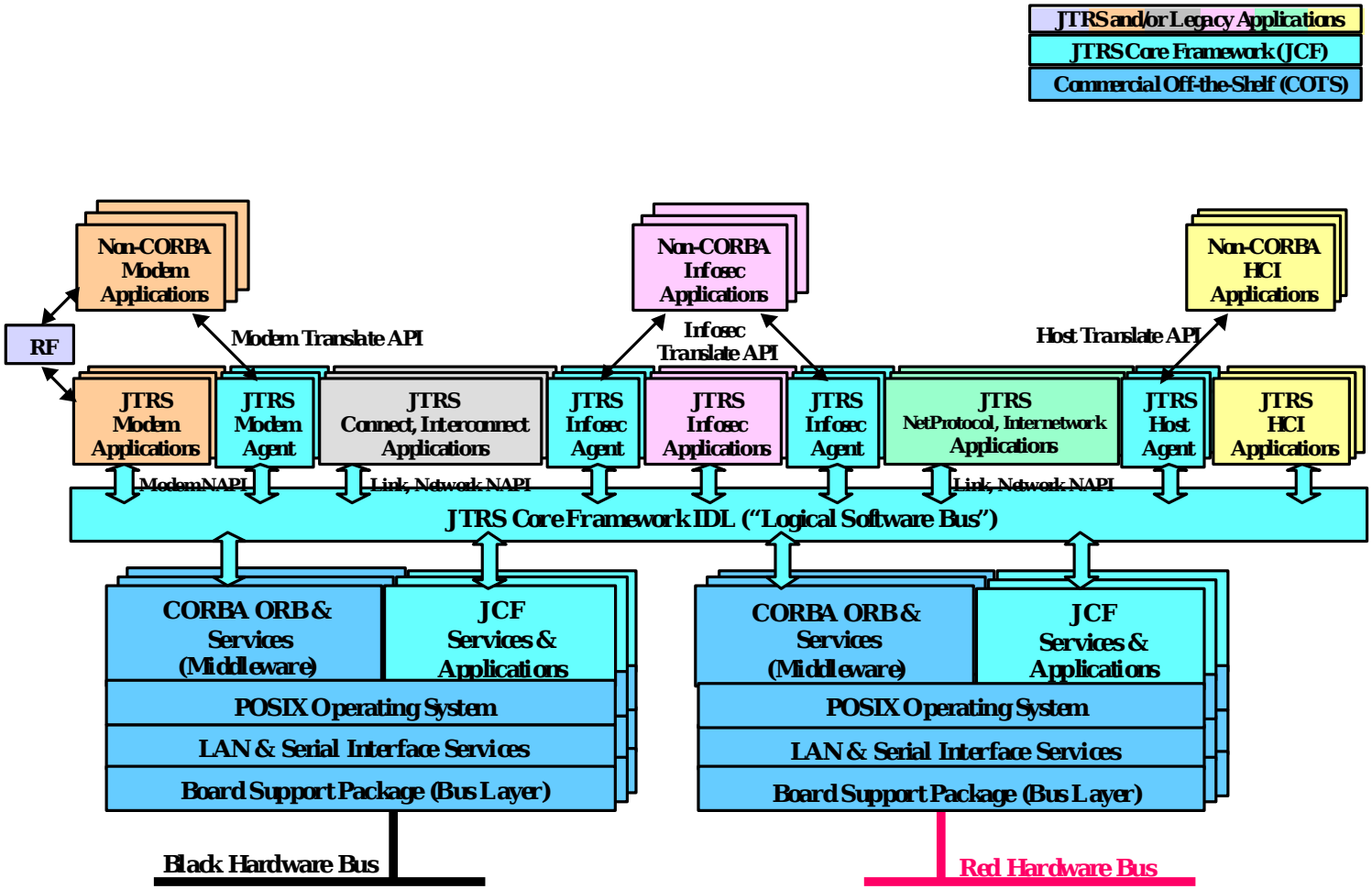
Step 1 Technical Evaluation Results

- **Where to Draw "The Line"?**
  - All Three Consortia Defined Their Architectures at Either The API Layer, or at the APIs Plus a Middleware Layer
- **All Three Consortia Focused Architecture Detail on the Software**
  - Level of Detail Shows a Good Start to an Architecture
- **Prior to Implementation (e.g., Building Prototypes), Need To:**
  - Define Standard Object/API Taxonomy, Semantics, and Performance Characteristics Necessary for Waveform Portability and Object Reuse
  - Precisely Define Required CORBAservices and POSIX Services
- **Waveform Portability**
  - At the Source-Code Level--Some Degree of Re-Porting Necessary to Move Between Implementations

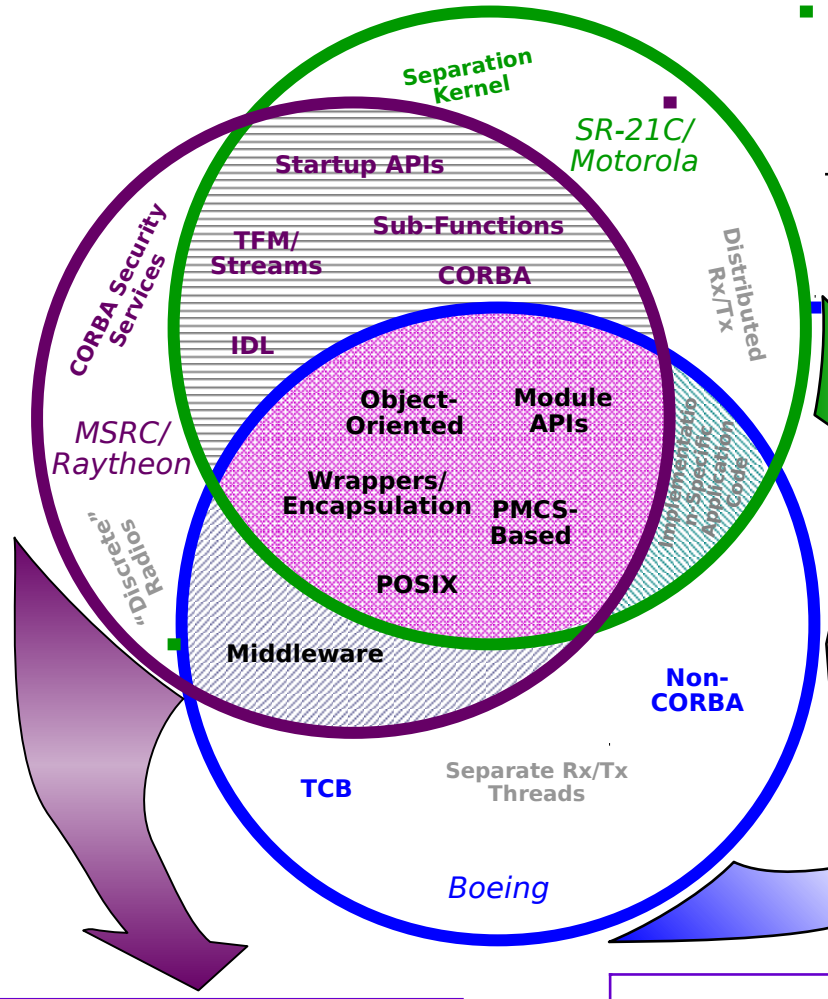
Step 1 Technical Evaluation Results

- **Summary**
  - Conceptually, All Three ADRs More Alike Than Different. Since Government Had to Pick ***One*** Architecture, the Technical Evaluation Team Recommended the MSRC Architecture Definition As the Framework for Step 2 Software Communications Architecture (SCA).
  - Architecture Guidelines Address Specific Areas For Further Emphasis During Step 2:
    - Standard APIs (Taxonomy, Semantics, and Performance Requirements)
    - Definition of Required CORBAservices and POSIX Services
    - Support/Development Environment
    - etc.
  - Several “Golden Nuggets” Gleaned From SR-21C and Boeing Consortia ADRs
    - Incorporated Into the Architecture Guidelines

Selected ADR SW Architecture



Step 1 SCA Outline



"Golden Nuggets"

- System Control
- Interface Descriptions
- Functional Entity Diagrams
- Cosite Mitigation
- Development Environment
- Software Distribution

Architecture Definition Report → **SCA Outline** ← **Architecture Guidelines**

Looking Ahead (To Step 2)

- **Step 2A Activities**
  - Complete the Architecture Development

Step 1 Architecture Definition Report

+

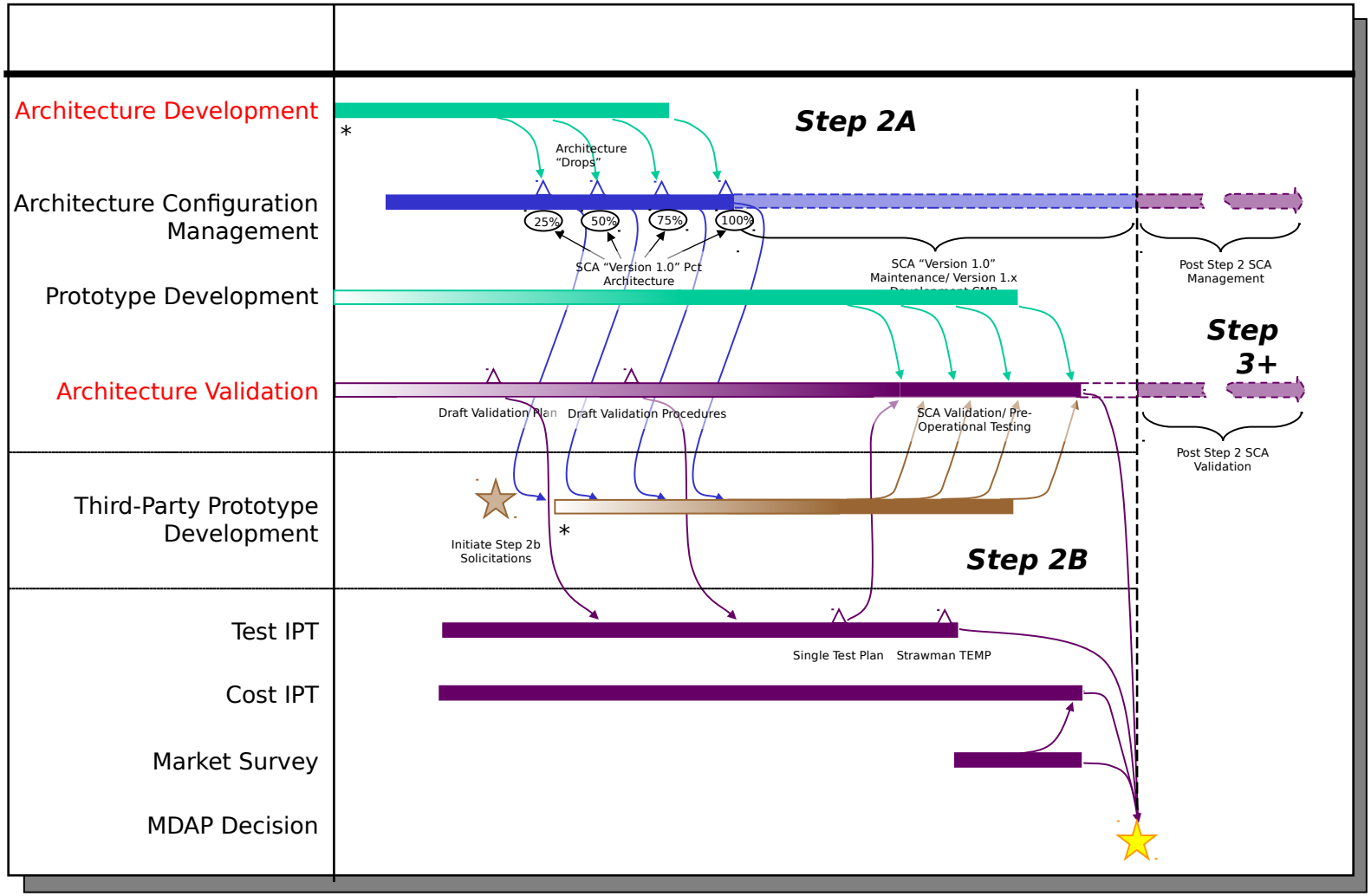
Architecture Guidelines

Software Communications Architecture (SCA)

Plus Work!

  - Validate the Architecture
    - Prototype Development
      - Architecturally-Compliant
      - Multiple Physical Implementations
      - Multiple Form Factors
    - Prototype Testing/Analysis
      - Includes Step 2B Prototypes
    - Software Portability
      - Between Different Vendors' Implementations

Step 2 Notional Schedule



* : Prototypes/ Waveforms Identified

Architecture Development

Architecture Development

- **Using Selected Framework as a Baseline, Develop an Open Architecture to the Point That It:**
  - Specifies a Structured Hardware and Software Framework
  - Defines the Hardware and Software Components From Which Future Communications Systems Are Built
  - Specifies Interactions Between the Components
  - Provides Patterns That Guide Component Composition
  - Provides Design Rules for the Components and Interconnections
- **Open Architecture**
  - The Architecture Specification Is Public
  - Developed to the Point That Third-Parties Can Produce Compliant Hardware And/Or Software

Architecture Development

- **The Step 1 Framework Is Not Complete**
  - The Architecture Guidelines Identify Areas of Concern, and Areas Requiring Further Definition
    - Issues Identified During the Step 1 Technical Evaluation
    - “Golden Nuggets” Mined From SR-21C and Boeing Consortia Reports Provide Examples of the Scope and Detail Desired.
- **When Is “Version 1.0” Complete?**
  - Guidelines Document Issues Have Been Addressed
  - Comments From Interim Releases Have Been Addressed

SCA Version 1.0 Is the Basis for Step 2 Validation Testing

Architecture Development

- **How Does Government Ensure Openness?**
  - Government Intends to Solicit Public Review and Comment on the Architecture As It Is Being Developed
    - Review of Interim “Drops” Will Aid in the Identification/ Resolution of Architecture Definition Issues
    - When Comments Have Been Addressed in a Mutually-Agreeable Manner, Step 2B and Validation Can Proceed

Architecture Development Key Areas

- **Modular Functional Breakout**
  - Sufficient Granularity to Allow For Innovation and Product Differentiation
- **Well-Defined Functional Entities Associated With the Functional Breakout**
  - Includes Functional Components for Each Defined Entity, And ***All*** the Interfaces for Functional Components Associated With Each Entity
- **Rule Sets for the Functional Entities, and Implementation Guidance**
  - Includes Adding New Functional Entities, and Functional Components to the Existing Entities

Architecture Development Key Areas

- **Standards**
  - POSIX, CORBA, etc.
- **Domain Independence**
  - Operating Systems
  - Data Structures for Information Flow
  - Set of Common Object Components and Interfaces – i.e, Encoders, Decoders, Modulators, etc.
- **System Control**
  - Process Startup, Teardown, etc.
- **Software Development and Distribution**
  - Waveform Development
  - Transfer/Distribution of Waveforms, Algorithms, Keys, etc.

Architecture Development Key Areas

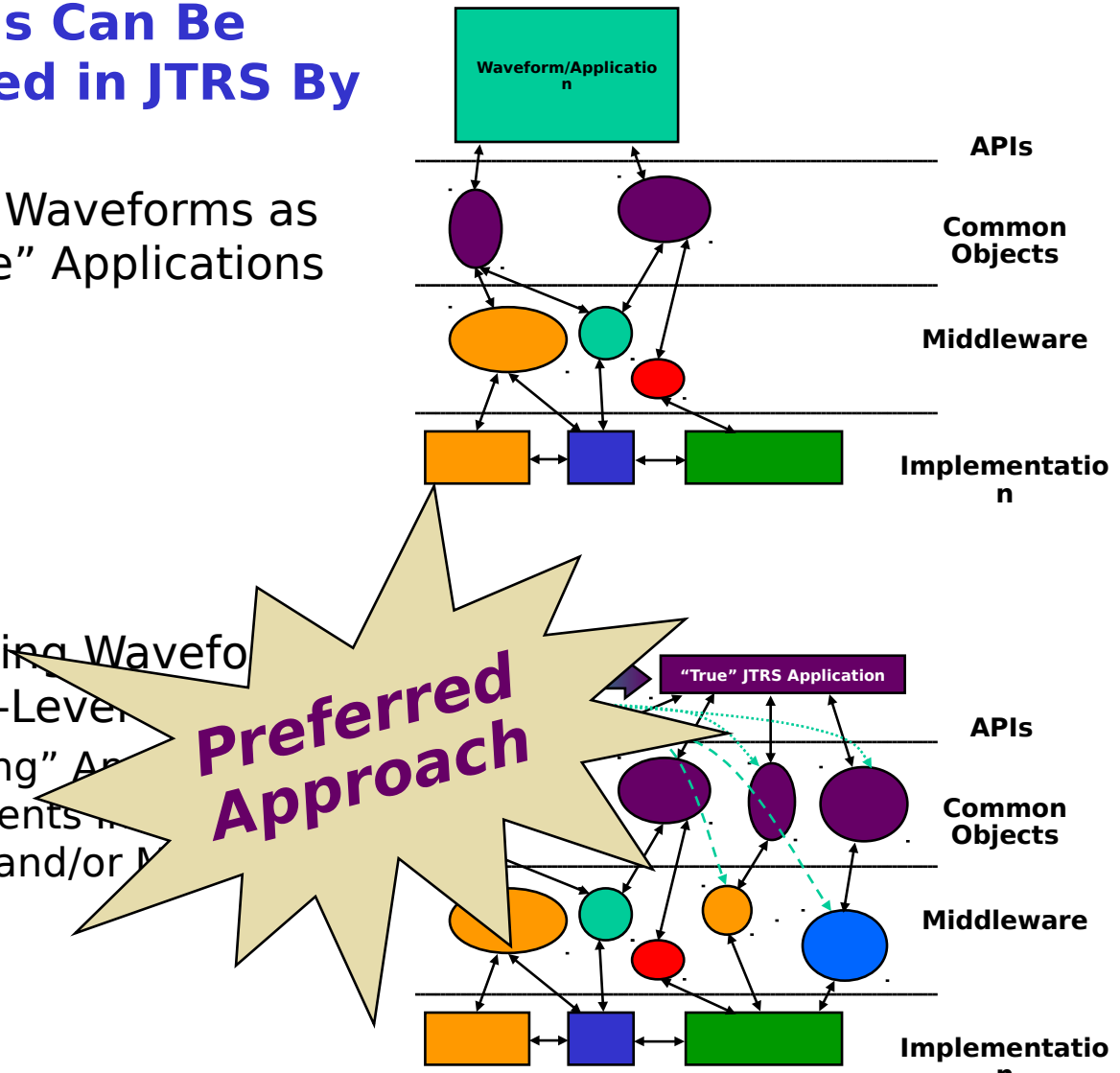
- **Interface Descriptions**
  - Development of a Standard Taxonomy of Objects/ APIs
    - Waveform Portability, and Independent Implementation, Requires Clearly-defined Semantics for Objects/APIs, As Well As Explicit Performance “Contracts”
  - Detailed Grammar for Expressing Parameters Used in APIs and Objects

Standard Object/API Taxonomy

- **Applications Can Be Implemented in JTRS By Either:**
  - Packaging Waveforms as "Aggregate" Applications

or...

  - Decomposing Wavefo… Into Lower-Level…
    - "Migrating" Ap… Components i… Objects and/or M…



Architecture Validation

Architecture Validation

- **What Does it Mean to "Validate" the Architecture?** –

  val•i•date \'val ə,dat \ (v) to corroborate or support on a sound basis

- **Initial SCA Validation is Complete When:**
  - Testing "Stresses" the Architecture With a Select Set of Waveforms and Cryptographic Algorithms/Functions Implemented on Multiple Architecturally-Compliant Prototypes
    - Differing Form Factors, Bus Structures, Processors, etc.
  - Demonstration and Testing Verifies Third-Party Ability to Implement Architecturally-Compliant Waveforms and Prototypes
  - Software Portability Between Independently-Developed Prototypes Has Been Tested
  - Analysis Indicates That the Architecture is Capable of Supporting All JTRS ORD Requirements

Prototypes

- **What is a Prototype?**

  pro•to•type \ˈprōd ’,tip\ (n) hardware and/or software implementation of the architecture

- **Highly-Desirable That Prototypes Provide:**
  - Multiple Different Physical Implementations
    - Demonstrate Multiple Methods of Implementing the Architecture to Accepted Standards
  - Domain- and Platform-Specific Implementations
    - Provide Basis for Service Procurement of JTRS
  - Waveform Applications for Step 2
    - Portable Between Prototypes
    - Stress the Architecture Across Frequency, Bandwidth, Data Rates, Hop Rates, and Modulation Modes
    - Legacy Radio System Interoperability

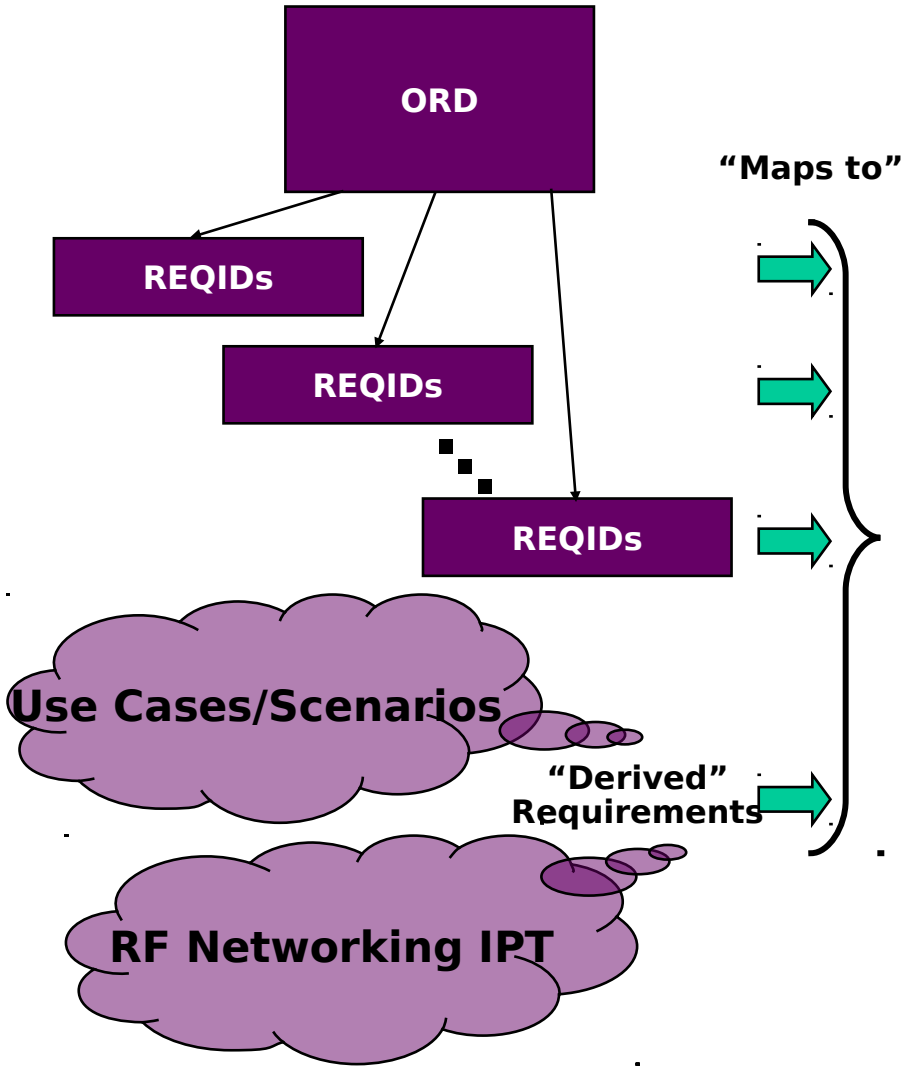
Architecture Validation Testing

- **An Integrated Test Plan is Required**
  - Built Upon Analysis, Demonstrations, and Testing of Architecturally-Compliant Prototypes
    - Must Address Compliance Testing of Any Prototypes
  - Meets Critical Aspects of the JTRS ORD and Program Goals
    - Scalable/modular/ Evolvable Architecture
    - Programmable/ Re-programmable/ Portable Software
    - Interoperable/ Waveforms
    - Crossbanding
    - Multiple Vendors
    - etc.

Possible Approach to Architecture Validation Testing

- **Who Writes the Test Plan?**
  - Contractor-Developed (With Participation of the Test IPT)
  - Government-Approved
- **How to Structure Test Plan?**
  - Map ORD/Derived Requirements to Multiple Prototypes
    - Demonstrating That the Architecture is “Scalable” Could be Accomplished Through a Mixture of Prototypes, Including, for Example, a Single Channel Handheld, a Dual-Channel Manpack, and a Four-Channel Maritime JTRS, Utilizing the Same Software Architecture and Waveforms
  - Allocate Validation of Individual Test Criteria to Appropriate Test Events
    - Some Testing Can Best be Performed at the Factory. Other Testing May Require Participation From the JPO, Service Labs, ???
  - “One Time Test”: Don’t Repeat Tests Unnecessarily

Possible Approach to Architecture Validation Testing



Verification Cross-Reference Matrix (VCRM)

REQID	ORD Ref	Level	Method	Event
T0001	4a(1)a	Component	Test	STI
T0002	4a(1)b	Subsystem	Test	STII
T0003	4a(1)c	System	Inspect	STIII
T0004	4a(1)d	Subsystem	Similarity	STII
. . .	. . .	. . .	. . .	. . .
T000*n*	7b	System	Analysis	STIII

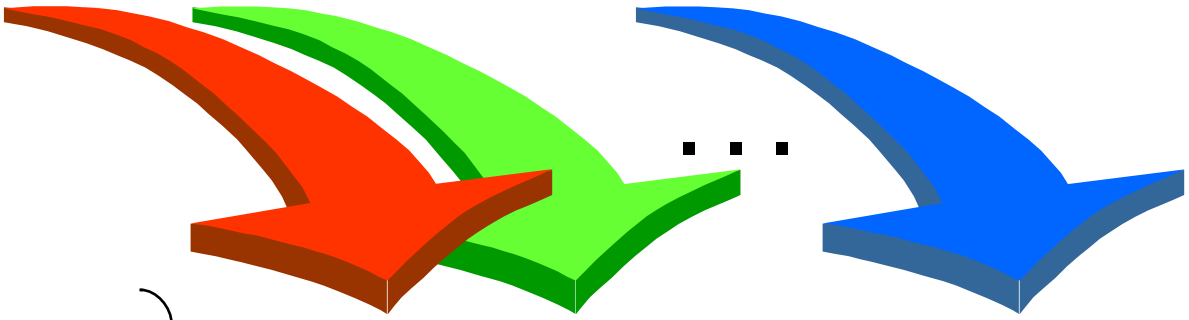
VCRM Used to Manage the Process of Architecture Validation

Possible Approach to Architecture Validation Testing

Verification Cross-Reference Matrix (VCRM)

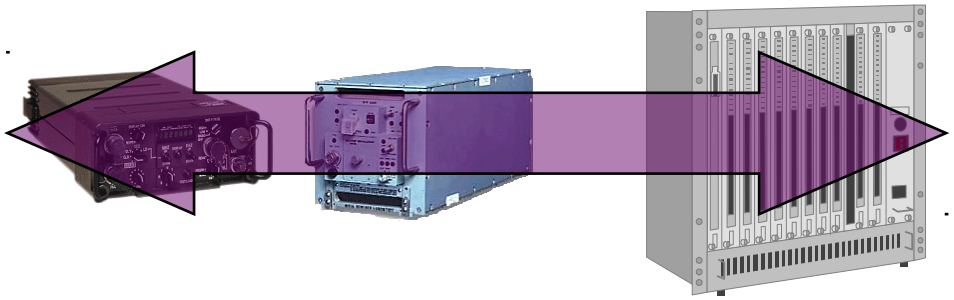
REQID	ORD Ref	Level	Method	Event
T0001	4a(1)a	Component	Test	STI
T0002	4a(1)b	Subsystem	Test	STII
T0003	4a(1)c	System	Inspect	STIII
T0004	4a(1)d	Subsystem	Similarity	STII
T000*n*	7b	System	Analysis	STIII

VCRM Elements Mapped to Multiple Prototypes



Prototypes/Waveforms Selected on Their Ability to "Test" Architecture Boundaries:

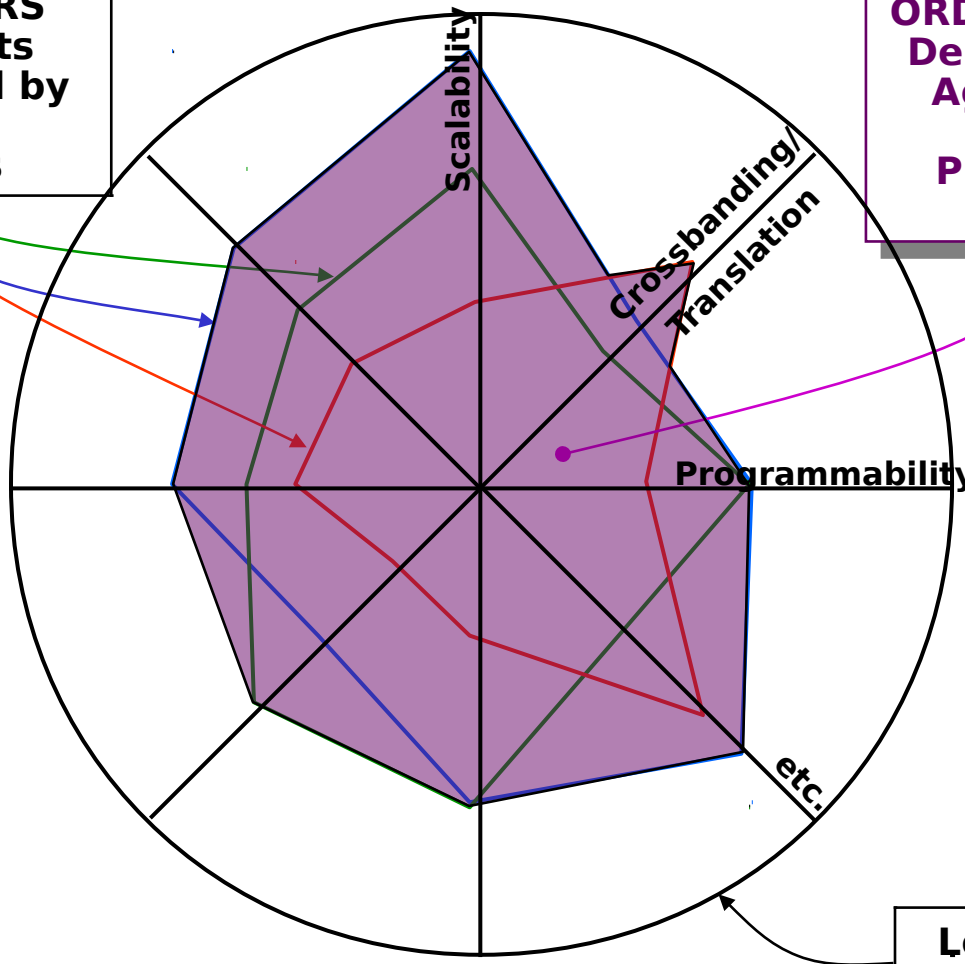
- **Scalability (Large/Small Form Factors)**
- **Interoperability**
- **Waveform Processing**
- **etc.**



Possible Approach to Architecture Validation

Portion of JTRS Requirements Demonstrated by Individual Prototypes

Subset of JTRS ORD Requirements Demonstrated by Aggregation of Individual Prototype Test Events



Loci of Total JTRS ORD Requirements

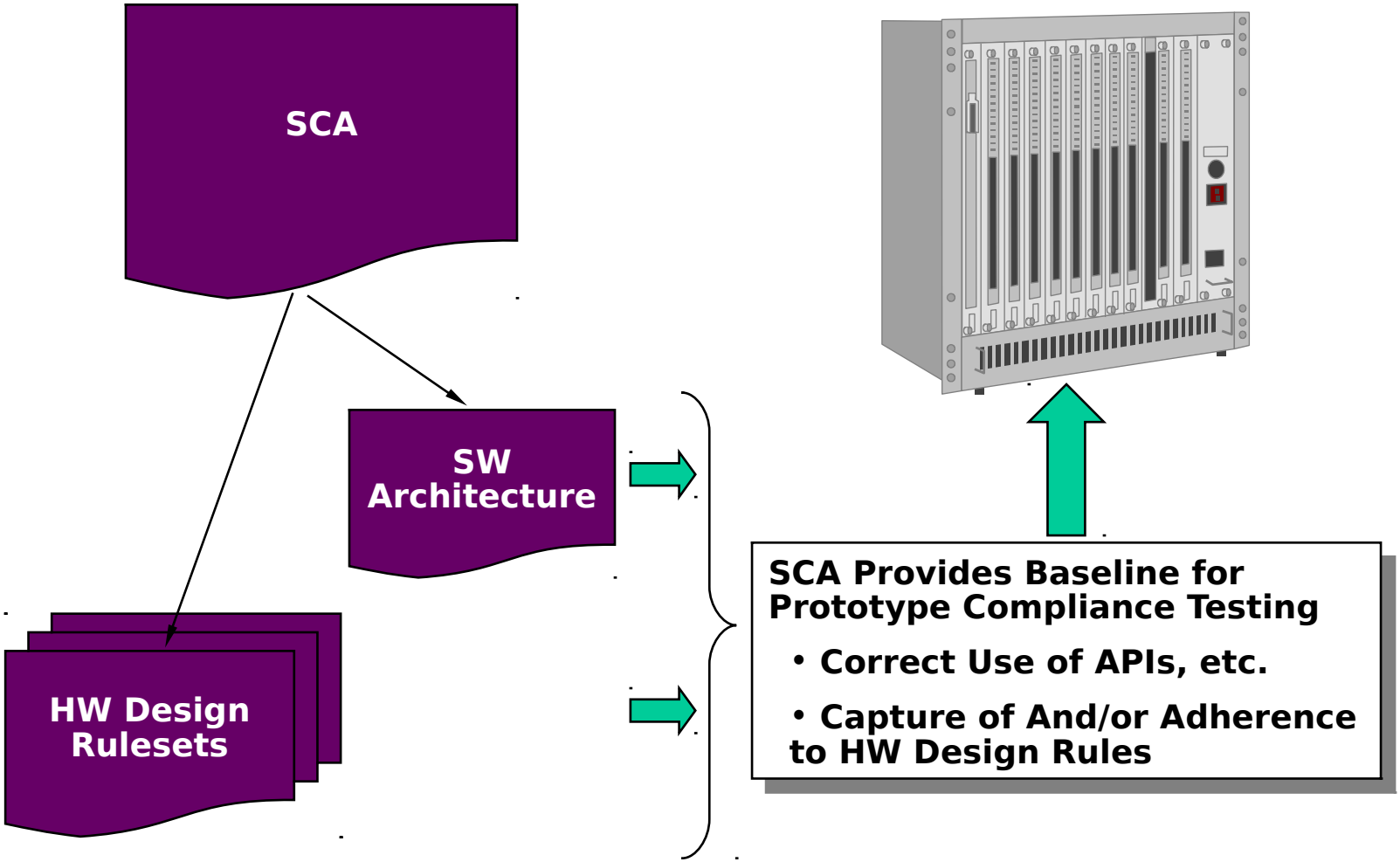
Compliance Testing

- **What Does "Compliance" Mean?**
  - A Prototype Correctly Implements the SCA
    - Underlying Software/ Hardware Implementation Supports Proper Execution of APIs/ Middleware Software Calls
    - External Interfaces Conform to Architectural Requirements
      - RF, LAN/WAN, etc.
- **Compliance Is Verified Through Over-the-Air and/or Lab Testing, Inspection, etc.**
  - Captures the Technical Descriptions of the Architecture
  - Measures Whether a Product Faithfully Implements the Architecture
- **Where to Accomplish This?**
  - Combination of Contractor/Government Facilities

Compliance Testing

- **Prototype Compliance Testing Provides Basis for the JTRS Support Environment (JSE)**
  - Accurate and Reliable Compliance Test Suites
  - Software Tools
  - Data Sets
  - Reference Material
  - etc.

Possible Approach to Compliance Testing


SCA
SW Architecture
HW Design Rulesets
SCA Provides Baseline for Prototype Compliance Testing
• Correct Use of APIs, etc.
• Capture of And/or Adherence to HW Design Rules

Summary

Summary

- **Two Major Goals of Step 2:**
  - Complete Development of the Architecture, and
  - Validate the Resulting Architecture
    - With
      - Compliant Prototypes
      - Compliant Waveforms
    - Using
      - Inspection
      - Analysis
      - Testing
      - Demonstration
- **Secondary Goal to Accelerate Development of Production Systems**